

Unidad Aritmética y Lógica (ALU).

Este bloque esta destinado a realizar las operaciones aritméticas y lógicas del microprocesador, es decir, es la unidad que hace todo el procesamiento. Esta unidad además de realizar las operaciones aritméticas y lógicas, también realiza el cálculo de las direcciones para el manejo de bloques de datos o arreglos en la memoria de datos.

La ALU es de 8 bits y esta implementada con un esquema de acarreo anticipado por generación y propagación para tener un tiempo de respuesta de 2 retardos de propagación de forma constante para la obtención de los acarreos. En la ilustración 1 se muestra el bloque de la ALU.

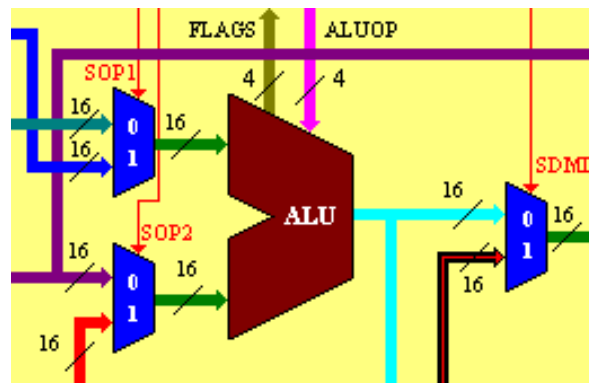


Ilustración 1 Unidad Aritmética y Lógica

Como se puede observar en la ilustración 1, la ALU maneja las siguientes señales:

- Dos **Buses de entrada de 8 bits**. Estos buses proporcionan los operandos a usar en las operaciones aritméticas y lógicas. Los operandos pueden provenir del archivo de registros, la pila o de la parte baja del formato de instrucción (I[11-0]).
- Un **Bus de salida de 8 bits**. En este bus se coloca el resultado de la operación aritmética o lógica.
- Un **Bus ALUOP de entrada de 4 bits**. Es un bus que permite seleccionar la operación aritmética y lógica a realizar por la ALU.
- Un **Bus de banderas de salida de 4 bits**. En este bus se mandan los valores de las banderas Z (zero), C (carry), N (negative) y OV (overflow) que genera cada operación en la ALU.

De cada operación se obtienen los valores de las banderas de acarreo (C), negativo (N), cero (Z) y desbordamiento (OV). Estas banderas son conocidas como **banderas de estado** puesto que proporcionan el estado de la ALU después de cada operación aritmética o lógica. Estas banderas son almacenadas en el registro de banderas o registro de estado para poder ser usadas con las instrucciones de comparación y brincos condicionales.

Cada una de las banderas de la ALU tiene las siguientes funciones:

- **Bandera Z (zero).** Esta bandera se pone en 1 cuando los 8 bits del resultado de la ALU son cero, en caso contrario tiene 0. Esto se logra con una compuerta NOR que tiene como entradas todos los bits de resultado.
- **Bandera C (carry).** Esta bandera muestra el valor que tiene el último acarreo de la ALU, en el caso de una ALU de 8 bits, el acarreo C8 es la bandera C.
- **Bandera N (negative).** Esta bandera muestra el valor que tiene el bit más significativo (bit de signo) del bus del resultado, en el caso de una ALU de 8 bits, el bit 7 de resultado es la bandera N. Cuando N tiene 1 significa que el resultado es negativo, de lo contrario es positivo.
- **Bandera OV (overflow).** Esta bandera es usada para operaciones aritméticas con signo. Cuando $OV = 1$ marca un desbordamiento en el resultado, lo que significa que el resultado no se puede representar con los bits que tenemos para el resultado. Esto se logra con una compuerta XOR entre los dos acarreos más significativos, en el caso de una ALU de 8 bits, la entrada de la compuerta XOR es C7 y C8.

Las operaciones aritméticas y lógicas que se pueden realizar dependen de la arquitectura diseñada de la ALU. En la ilustración 2, podemos observar la arquitectura para un bit del ALU del ESCOMIPS.

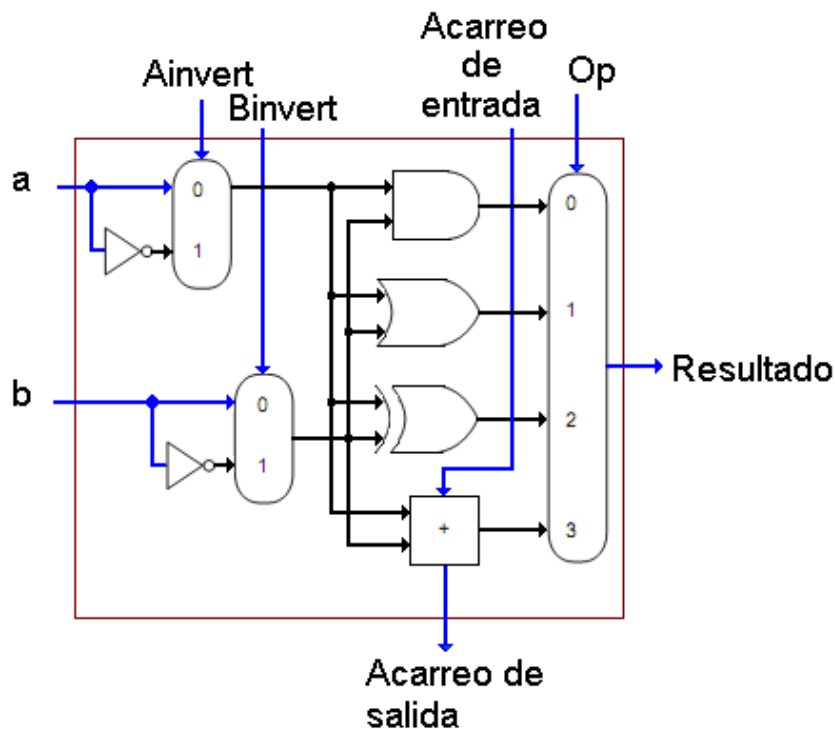


Ilustración 2 ALU de un bit



Las operaciones básicas que se pueden realizar con la arquitectura elegida son:

Ainvert ALUOP(3)	Binvert ALUOP(2)	Op(1) ALUOP(1)	Op(0) ALUOP(0)	Operación	Banderas que afecta la operación
0	0	0	0	Operación lógica AND $Resultado = a \cdot b$	N Z
0	0	0	1	Operación lógica OR $Resultado = a + b$	N Z
0	0	1	0	Operación lógica XOR $Resultado = a \oplus b$ $= a \cdot \bar{b} + \bar{a} \cdot b$	N Z
1	1	1	0	Operación lógica XOR $Resultado = \bar{a} \oplus \bar{b}$ $= \bar{a} \cdot \bar{b} + a \cdot b = \bar{a} \cdot b + a \cdot \bar{b}$	N Z
0	0	1	1	Operación aritmética $Resultado = a + b$	N Z C OV
0	1	1	1	Operación aritmética $Resultado = a - b$	N Z C OV
0	1	1	0	Operación lógica XNOR $Resultado = a \oplus \bar{b} = a \cdot \bar{b} + \bar{a} \cdot b$ $= a \cdot b + \bar{a} \cdot \bar{b} = \overline{a \oplus b}$	N Z
1	0	1	0	Operación lógica XNOR $Resultado = \bar{a} \oplus b = \bar{a} \cdot b + a \cdot \bar{b}$ $= \bar{a} \cdot \bar{b} + a \cdot b = \overline{\bar{a} \oplus \bar{b}}$	N Z
1	1	0	1	Operación lógica NAND $Resultado = \overline{a + b} = \bar{a} \cdot \bar{b}$ Teorema de DeMorgan	N Z
1	1	0	0	Operación lógica NOR $Resultado = \overline{a \cdot b} = \bar{a} + \bar{b}$ Teorema de DeMorgan	N Z
1	1	0	1	Operación lógica NOT usando la compuerta universal NAND. si $a = b$ $Resultado = \bar{a} + \bar{a} = \bar{a} \cdot \bar{a} = \bar{a}$	N Z
1	1	0	0	Operación lógica NOT usando la compuerta universal NOR. si $a = b$ $Resultado = \bar{a} \cdot \bar{a} = \bar{a} + \bar{a} = \bar{a}$	N Z

La ilustración 3 muestra una ALU de 4 bits con las banderas de estado.

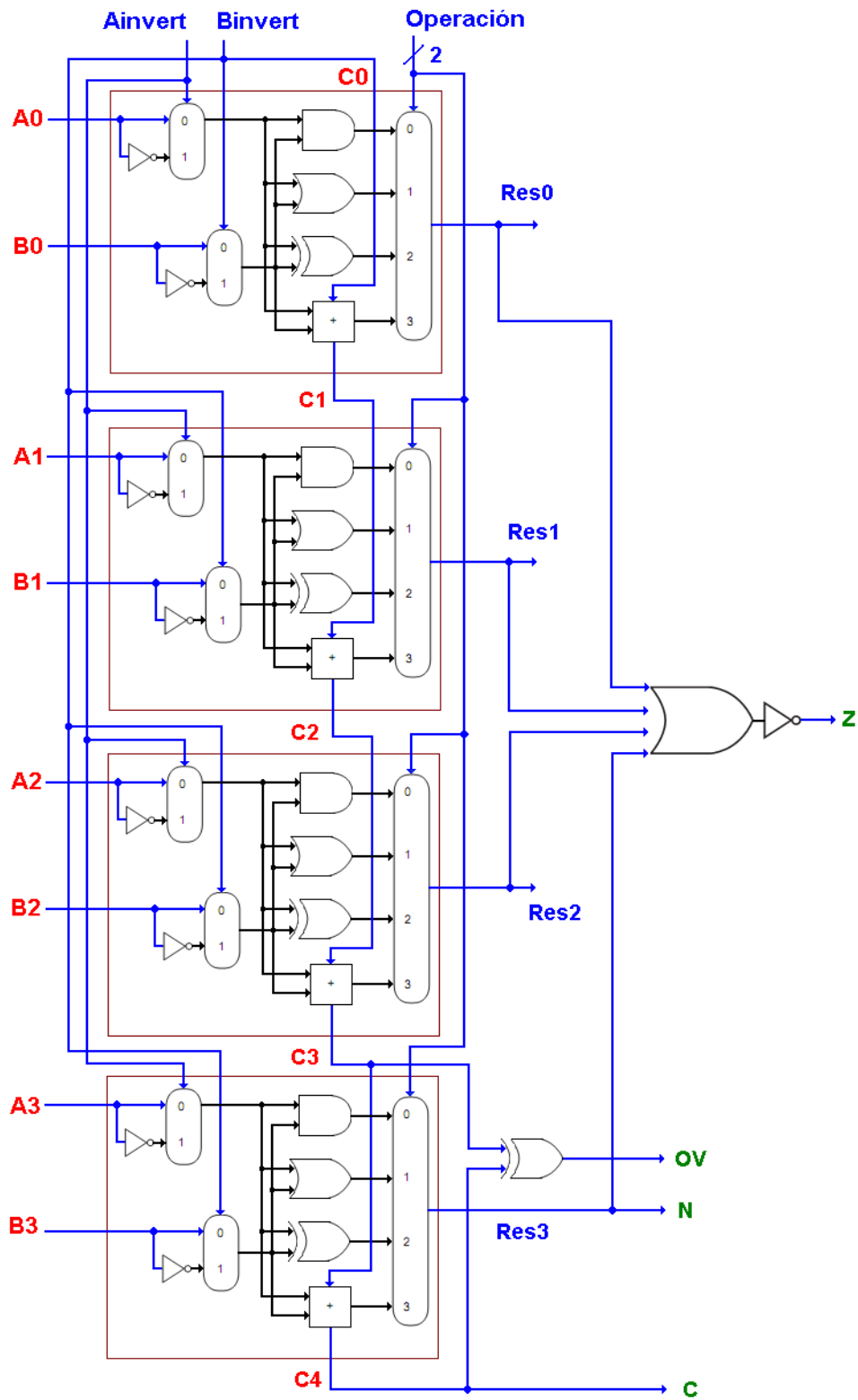


Ilustración 3: ALU de 4 bits.



Para verificar el correcto funcionamiento de la ALU vamos a usar valores de 4 bits, los cuales se muestran en la tabla 1.

A	B	C	D	SIN SIGNO	CON SIGNO
0	0	0	0	0	0
0	0	0	1	1	1
0	0	1	0	2	2
0	0	1	1	3	3
0	1	0	0	4	4
0	1	0	1	5	5
0	1	1	0	6	6
0	1	1	1	7	7
1	0	0	0	8	-8
1	0	0	1	9	-7
1	0	1	0	10	-6
1	0	1	1	11	-5
1	1	0	0	12	-4
1	1	0	1	13	-3
1	1	1	0	14	-2
1	1	1	1	15	-1

Tabla 1: Representación de números de 4 bits.

Los valores de prueba para la ALU se muestran a continuación:

OPERACIÓN AND				OPERANDOS					BANDERAS			
ALUOP				A = 5 =	A(3) 0	A(2) 1	A(1) 0	A(0) 1	OV	N	Z	C
AINVERT	BINVERT	OP(1)	OP(0)	B = -2 =	B(3) 1	B(2) 1	B(1) 1	B(0) 0				
0	0	0	0	RESULTADO = 4 =	0	1	0	0	0	0	0	0

OPERACIÓN OR				OPERANDOS				BANDERAS				
ALUOP				A = 5 =	A(3) 0	A(2) 1	A(1) 0	A(0) 1	OV	N	Z	C
AINVERT	BINVERT	OP(1)	OP(0)	B = -2 =	B(3) 1	B(2) 1	B(1) 1	B(0) 0				
0	0	0	1	RESULTADO = -1 =	1	1	1	1	0	1	0	0



OPERACIÓN XOR				OPERANDOS				BANDERAS				
ALUOP				A = 5 =	A(3) 0	A(2) 1	A(1) 0	A(0) 1	OV	N	Z	C
AINVERT	BINVERT	OP(1)	OP(0)	B = -2 =	B(3) 1	B(2) 1	B(1) 1	B(0) 0				
0	0	1	0	RESULTADO = -5 =	1	0	1	1				

OPERACIÓN NAND				OPERANDOS				BANDERAS				
ALUOP				A = 5 =	A(3) 0	A(2) 1	A(1) 0	A(0) 1	OV	N	Z	C
AINVERT	BINVERT	OP(1)	OP(0)	B = -2 =	B(3) 1	B(2) 1	B(1) 1	B(0) 0				
1	1	0	1	RESULTADO = -5 =	1	0	1	1				

OPERACIÓN NOR				OPERANDOS				BANDERAS				
ALUOP				A = 5 =	A(3) 0	A(2) 1	A(1) 0	A(0) 1	OV	N	Z	C
AINVERT	BINVERT	OP(1)	OP(0)	B = -2 =	B(3) 1	B(2) 1	B(1) 1	B(0) 0				
1	1	0	0	RESULTADO = 0 =	0	0	0	0				

OPERACIÓN XNOR				OPERANDOS				BANDERAS				
ALUOP				A = 5 =	A(3) 0	A(2) 1	A(1) 0	A(0) 1	OV	N	Z	C
AINVERT	BINVERT	OP(1)	OP(0)	B = -2 =	B(3) 1	B(2) 1	B(1) 1	B(0) 0				
1	0	1	0	RESULTADO = 4 =	0	1	0	0				

OPERACIÓN SUMA				OPERANDOS				BANDERAS				
ALUOP				A = 5 =	A(3) 0	A(2) 1	A(1) 0	A(0) 1	OV	N	Z	C
AINVERT	BINVERT	OP(1)	OP(0)	B = -2 =	B(3) 1	B(2) 1	B(1) 1	B(0) 0				
0	0	1	1	RESULTADO = 3 =	0	0	1	1				



OPERACIÓN SUMA				OPERANDOS				BANDERAS				
ALUOP				A = 5 =	A(3) 0	A(2) 1	A(1) 0	A(0) 1	OV	N	Z	C
AINVERT	BINVERT	OP(1)	OP(0)	B = 7 =	B(3) 0	B(2) 1	B(1) 1	B(0) 1				
0	0	1	1	RESULTADO = -4 =	1	1	0	0				

OPERACIÓN RESTA				OPERANDOS				BANDERAS				
ALUOP				A = 5 =	A(3) 0	A(2) 1	A(1) 0	A(0) 1	OV	N	Z	C
AINVERT	BINVERT	OP(1)	OP(0)	B = -2 =	B(3) 1	B(2) 1	B(1) 1	B(0) 0				
0	1	1	1	RESULTADO = 7 =	0	1	1	1				

OPERACIÓN RESTA				OPERANDOS				BANDERAS				
ALUOP				A = 5 =	A(3) 0	A(2) 1	A(1) 0	A(0) 1	OV	N	Z	C
AINVERT	BINVERT	OP(1)	OP(0)	B = 5 =	B(3) 0	B(2) 1	B(1) 0	B(0) 1				
0	1	1	1	RESULTADO = 0 =	0	0	0	0				

OPERACIÓN NAND (NOT)				OPERANDOS				BANDERAS				
ALUOP				A = 5 =	A(3) 0	A(2) 1	A(1) 0	A(0) 1	OV	N	Z	C
AINVERT	BINVERT	OP(1)	OP(0)	B = 5 =	B(3) 0	B(2) 1	B(1) 0	B(0) 1				
1	1	0	1	RESULTADO = -6 =	1	0	1	0	0	1	0	0